Freshness and Reactivity Analysis in Globally Asynchronous Locally Time-Triggered Systems

Frédéric Boniol[1], Michaël Lauer[2], Claire Pagetti[1], Jérôme Ermont[3]

(kindly presented by Pierre-Loïc Garoche)

[1]ONERA - Toulouse, France

[2]Ecole Polytechnique de Montréal, Canada

[3]IRIT - Toulouse, France

2013, May 13th-16th

Brief introduction and main contributions

Context: real-time embedded systems (e.g. avionic systems)

- built as functional chains from sensors to actuators through real-time tasks
- critical

$\Rightarrow$ need of formal verification methods

$\Rightarrow$ Main contributions:

- a verification method for end-to-end freshness and reactivity properties along functional chains
- via a Mixed Integer Linear Programming formalization
- scalable approach

Outline

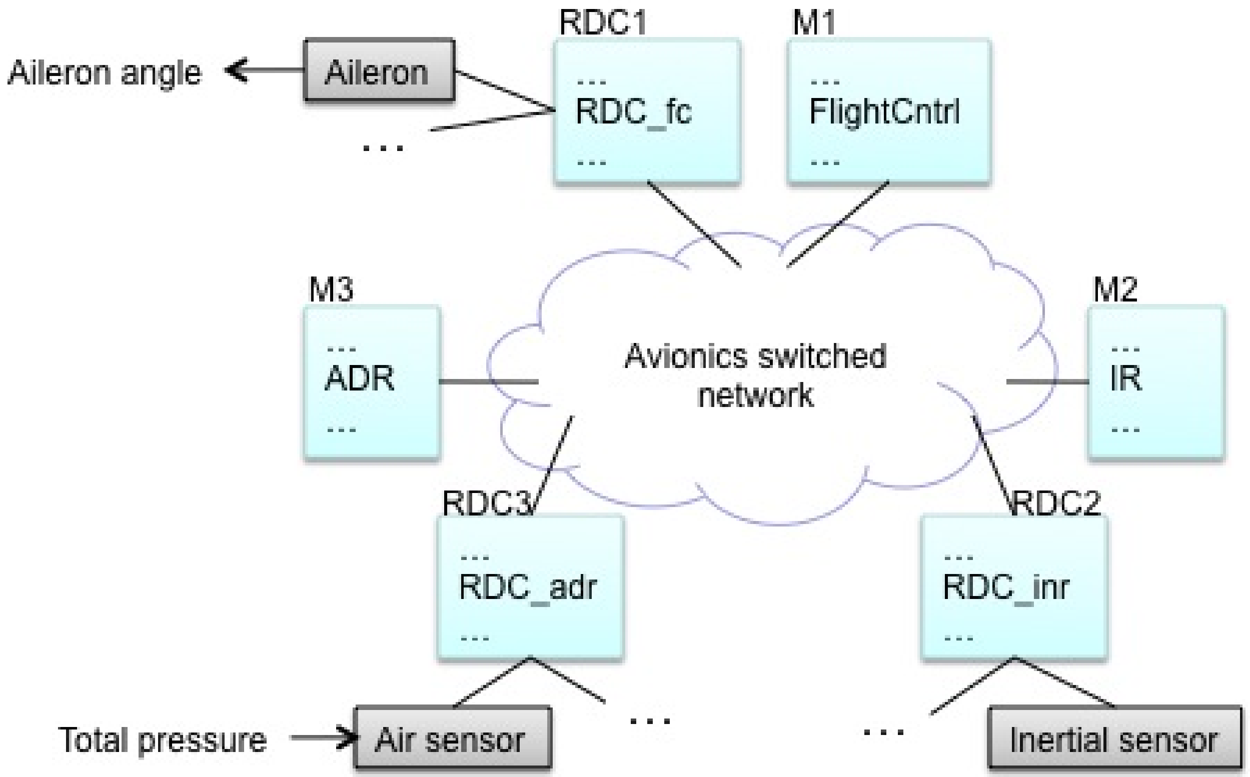
1. Context: real-time embedded systems
   - An avionic case-study
   - Generalization: models and hypotheses
2. Related approaches
   - Previous work
   - Contribution v.s. previous work
3. The freshness analysis method
   - General idea
   - Modeling
   - Results on the case study
4. Extension to reactivity requirements
5. Conclusion and next work

Outline

1. Context: real-time embedded systems
   - An avionic case-study
   - Generalization: models and hypotheses
2. Related approaches
   - Previous work
   - Contribution v.s. previous work
3. The freshness analysis method
   - General idea
   - Modeling
   - Results on the case study
4. Extension to reactivity requirements
5. Conclusion and next work

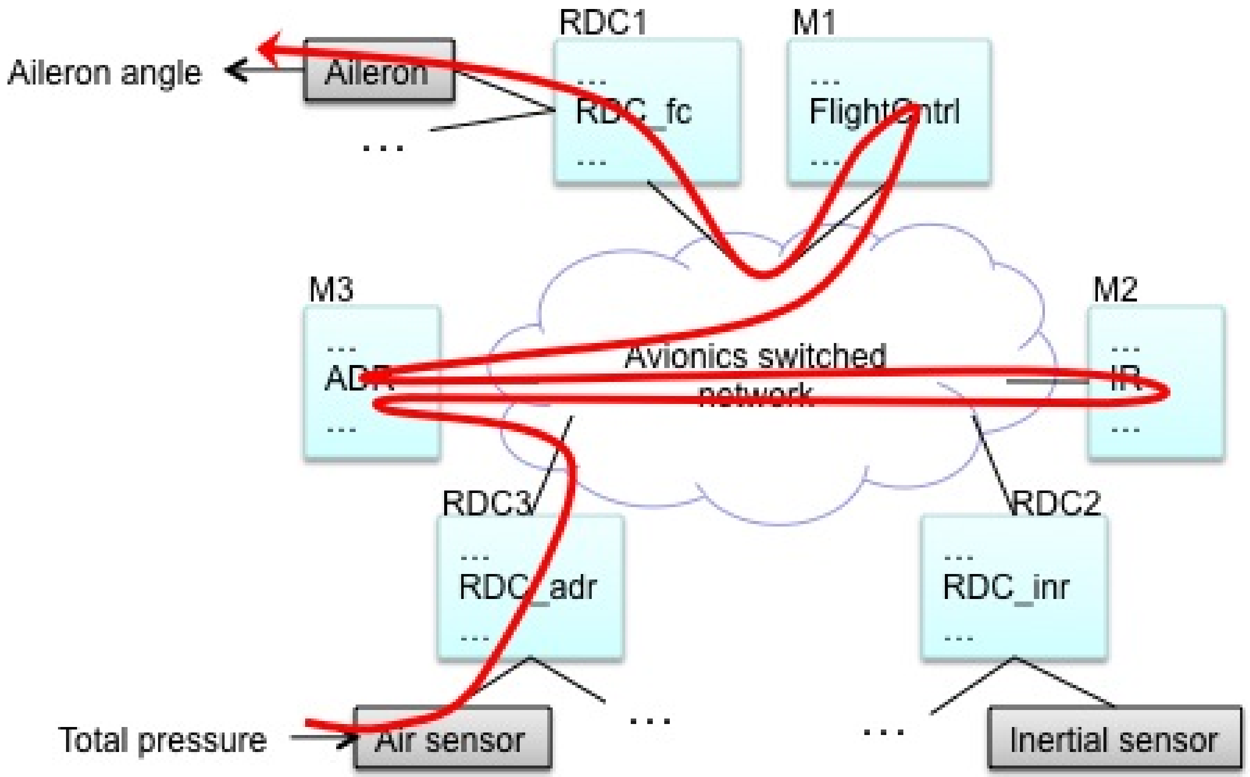
Outline

A Flight Control Chain (extract)



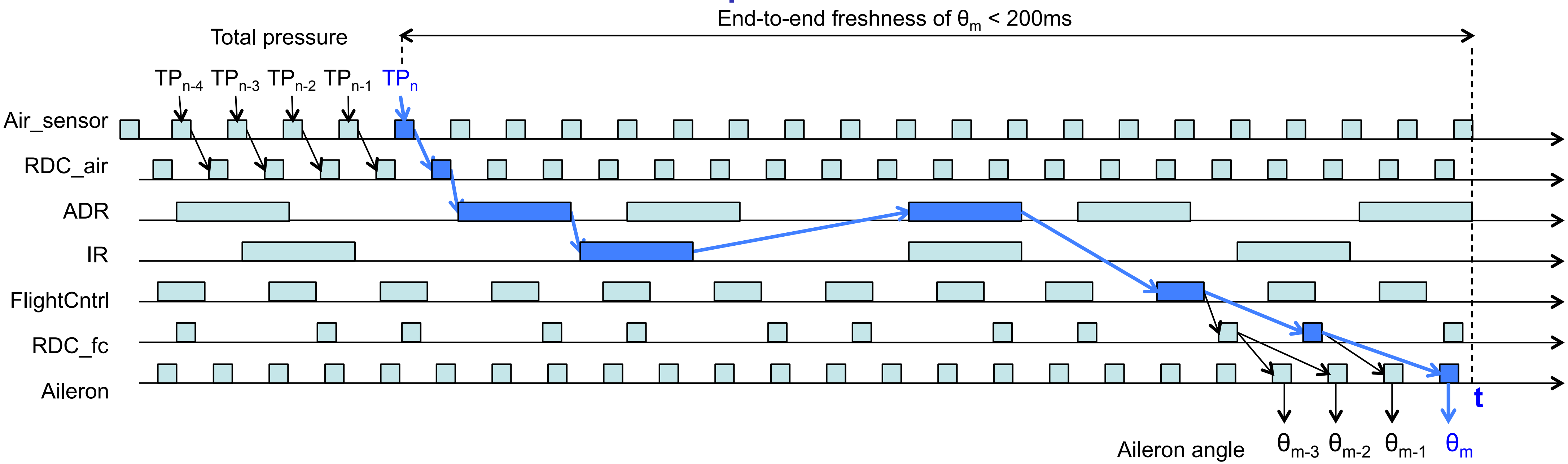
- computes the aileron angle w.r.t the current total pressure
- composed of 1 sensor, 1 actuator and 5 tasks mapped onto 5 processing modules
- tasks communicate through the avionics network

A Flight Control Chain (extract)



- computes the aileron angle w.r.t the current total pressure
- composed of 1 sensor, 1 actuator and 5 tasks mapped onto 5 processing modules
- tasks communicate through the avionics network

End-to-end freshness requirement



Requirement:

- for any date t, let θ_m the current aileron angle (at t),
- let TP_n the sample on which θ_m depends
- then freshness requirement: $t - date(TP_n) < 200ms$

Freshness

$F(t) = t - date(TP_n)$ is the freshness of the θ_m at date t: the age of the output with respect to its related input

Outline

1. Context: real-time embedded systems
   - An avionic case-study
   - Generalization: models and hypotheses
2. Related approaches
   - Previous work
   - Contribution v.s. previous work
3. The freshness analysis method
   - General idea
   - Modeling
   - Results on the case study
4. Extension to reactivity requirements
5. Conclusion and next work

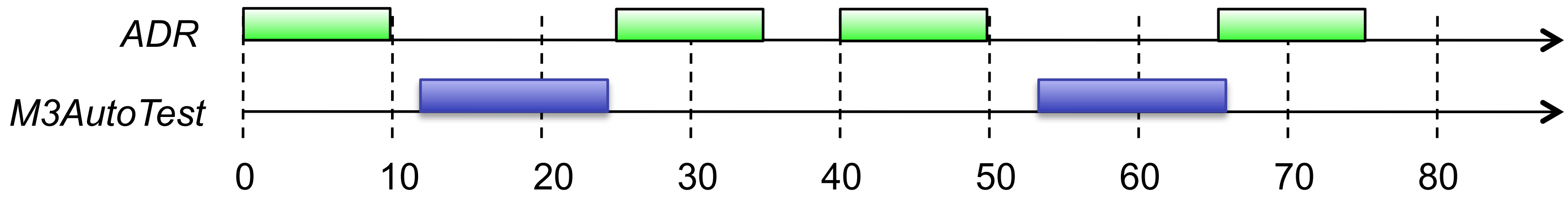
Model and hypotheses

System model:

- a set of periodic tasks $\Gamma = \{\tau_1, \ldots, \tau_N\}$
- statically mapped onto a set of modules $\mathcal{M} = \{M_1, \ldots, M_m\}$
- H_i is the hyper-period of the tasks mapped onto M_i.

Task model:

- each task τ_j on each module is characterized by a set of jobs $\{\tau_j(k)\}_{k=0 \ldots n_j}$ in the hyper-period of the module
- each job $\tau_j(k)$ is characterized by a timed interval $[b_j(k), e_j(k)]$ (static scheduling)



(example of module M_3)

Model and hypotheses

Communication model: tasks are assumed to communicate in an asynchronous way

- inputs are read at the beginning of the task
- outputs are produced at any time before the end of the task

Communication means:

- tasks running on the same module communicate through memory without delay
- tasks running on different modules communicate through a global network with bounded traversal times

Model and hypotheses

Global asynchronism: processing modules are globally asynchronous

- they can be shifted by an arbitrary amount of time.

$\Rightarrow$ Question: how to calculate the Worst Case Freshness (WCF) along a functional chain?

Outline

Outline

Previous work

Local approaches

- The holistic approach (Tindell et al. (1994), Spuri (1996), . . .)
- Real-Time Calculus (Thiele et al. (2000))

$\Rightarrow$ Can be pessimistic since they consider worst case scenario locally on every component along the functional chain

$\Rightarrow$ Lead to impossible scenarios

Network analysis methods

- Network Calculus (Le Boudec et al. (2001))
- Trajectory Approach (Martin et al. (2006))

$\Rightarrow$ Only for network traversal time

Previous work

Global approaches

- Global modeling by timed automata and model checking (Carcenac et al. (2006), Ning Ge et al. (2012))

$\Rightarrow$ higher expressive power (more general temporal properties)

$\Rightarrow$ however, suffer from the combinatorial explosion

$\Rightarrow$ then, not efficient enough for realistic systems.

Outline

The contribution v.s. the similar approaches

- to focus on end-to-end freshness / reactivity properties
- to propose a global encoding as an Mixed Integer Linear Program

$\Rightarrow$ less pessimistic than local approaches

$\Rightarrow$ more scalable than model checking

Outline

Outline

1. Context: real-time embedded systems
   - An avionic case-study
   - Generalization: models and hypotheses
2. Related approaches
   - Previous work
   - Contribution v.s. previous work
3. The freshness analysis method
   - General idea
   - Modeling
   - Results on the case study
4. Extension to reactivity requirements
5. Conclusion and next work

General idea

Principle

- to characterize all the possible behaviors of the functional chain with a set of constraints,
- to determine the worst case scenario among all these possible behaviors maximizing the freshness criteria
- can be done automatically by a solver.

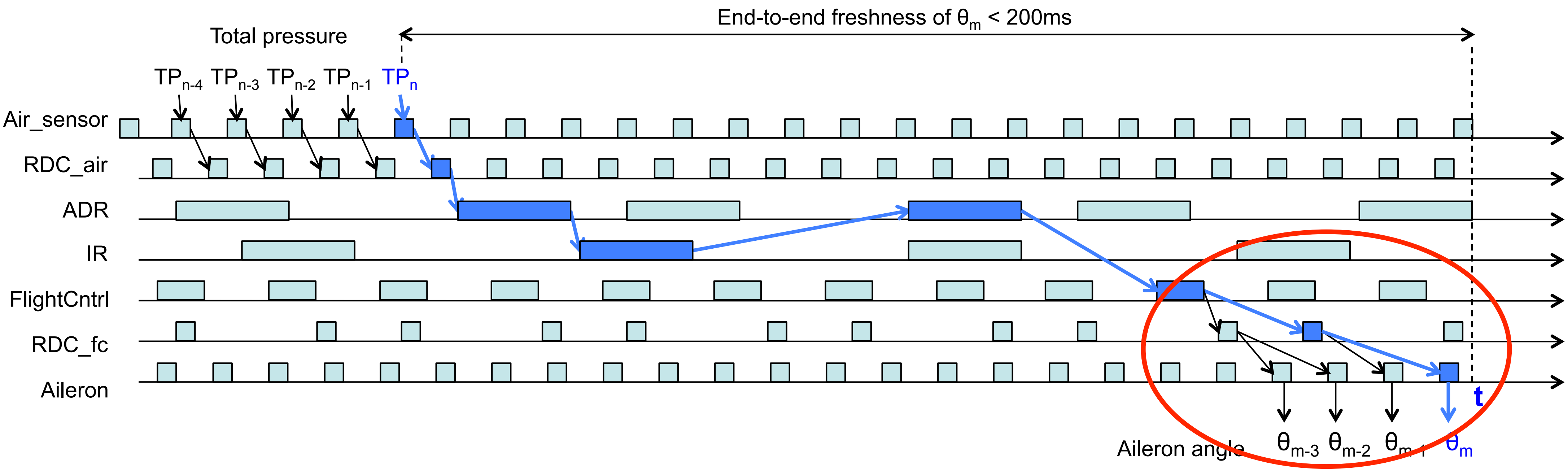
$\Rightarrow$ Challenge:

- to identify the variables of the modeling, and then the constraints defining accurately the behavior of the system
- in a scalable way

Outline

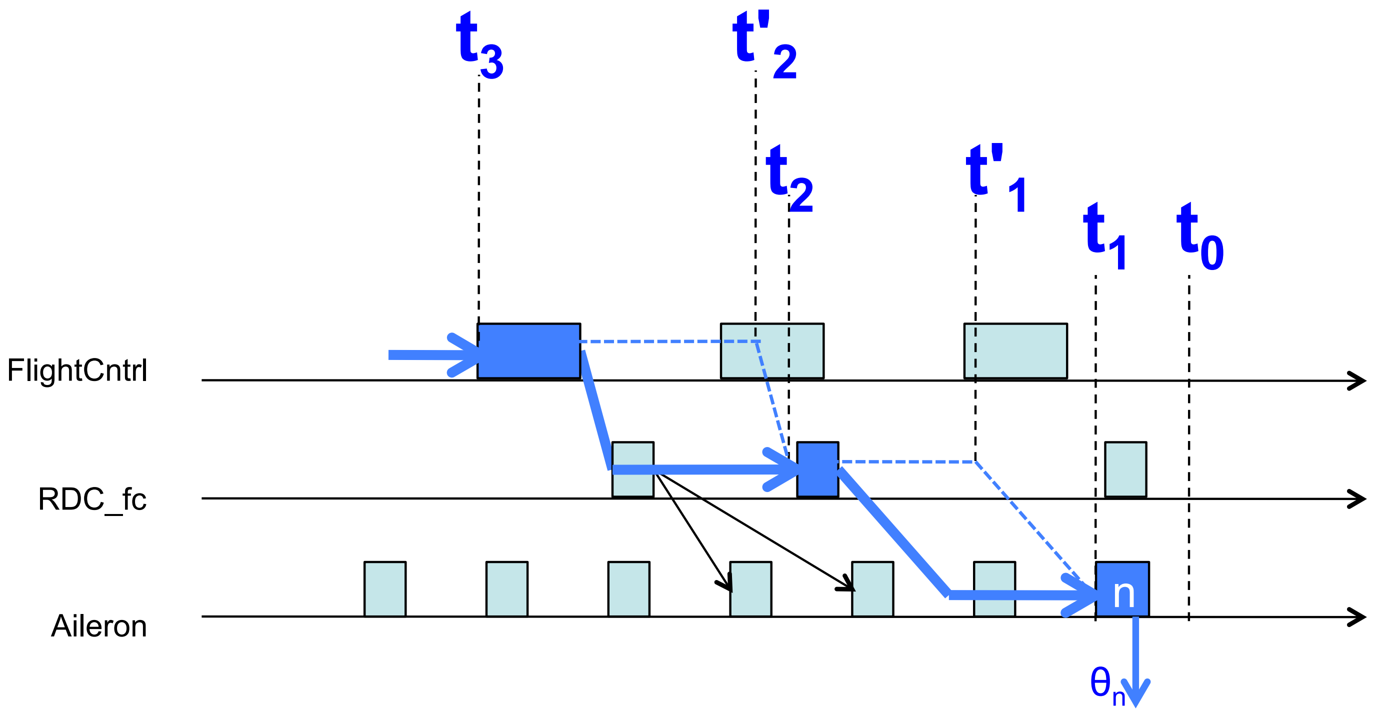
1. Context: real-time embedded systems
   - An avionic case-study
   - Generalization: models and hypotheses
2. Related approaches
   - Previous work
   - Contribution v.s. previous work
3. The freshness analysis method
   - General idea
   - Modeling
   - Results on the case study
4. Extension to reactivity requirements
5. Conclusion and next work

Case study recall



The modeling begins at the end
(freshness characterizes the output of the chain w.r.t. to the input)
$\Rightarrow$ Focus on the end of the chain

Case study recall

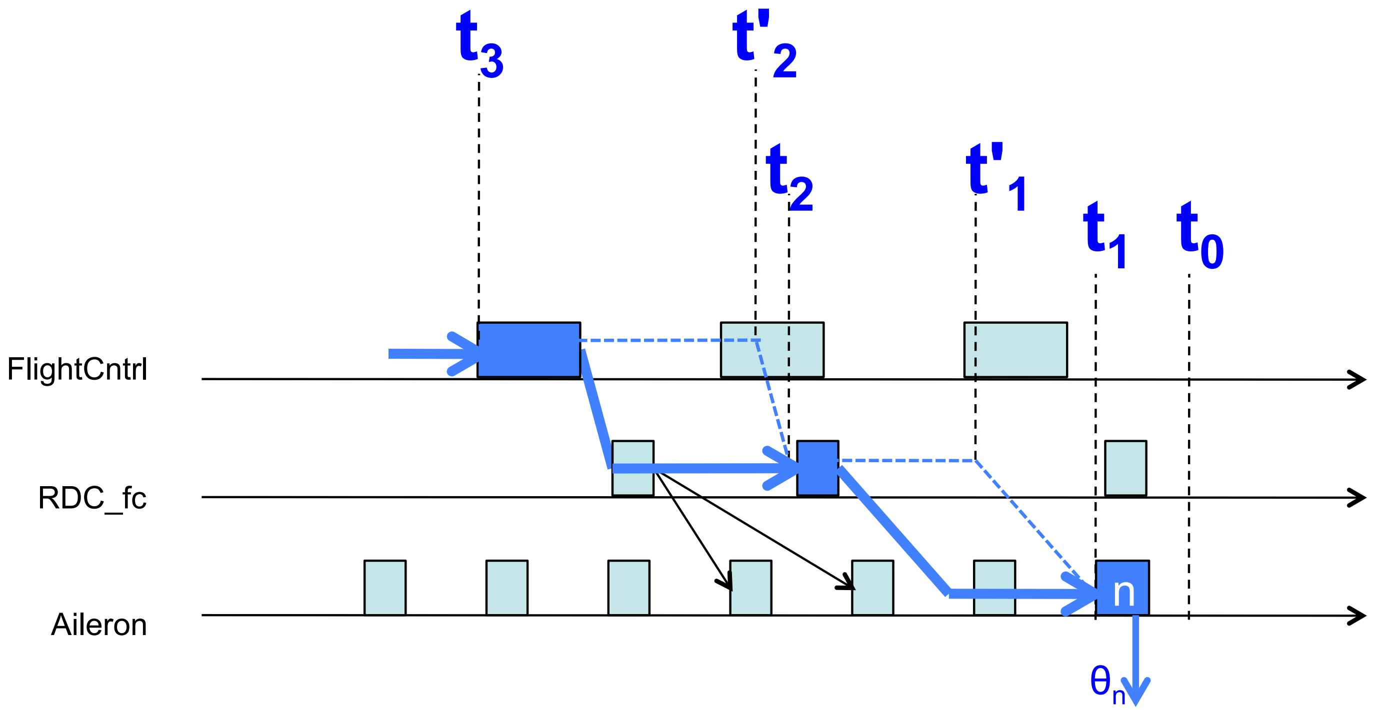


- t_0 = (arbitrary) date at which the output is observed
- n = occurrence of task "*Aileron*" which produces the output at t_0
- t_1 = date at which "*Aileron*[n]" begins and reads its input
- $t_1 - t'_1$ = communication time from "*RDC_fc*" to "*Aileron*"

$\Rightarrow$ t'_1 = date at which the output of "*RDC_fc*" is observed

- and so on...

Case study recall



$\Rightarrow$ Question: which constraints to characterize n, t_1, t_1', $t_2 \ldots$ from t_0?

Module (including sensors and actuators) modeling

- A module M_i is only characterized by an offset O_i
- Modules, sensors and actuators are asynchronous
- $\Rightarrow$ each O_i can be arbitrarily chosen

$\Rightarrow$ Constraints (case-study):

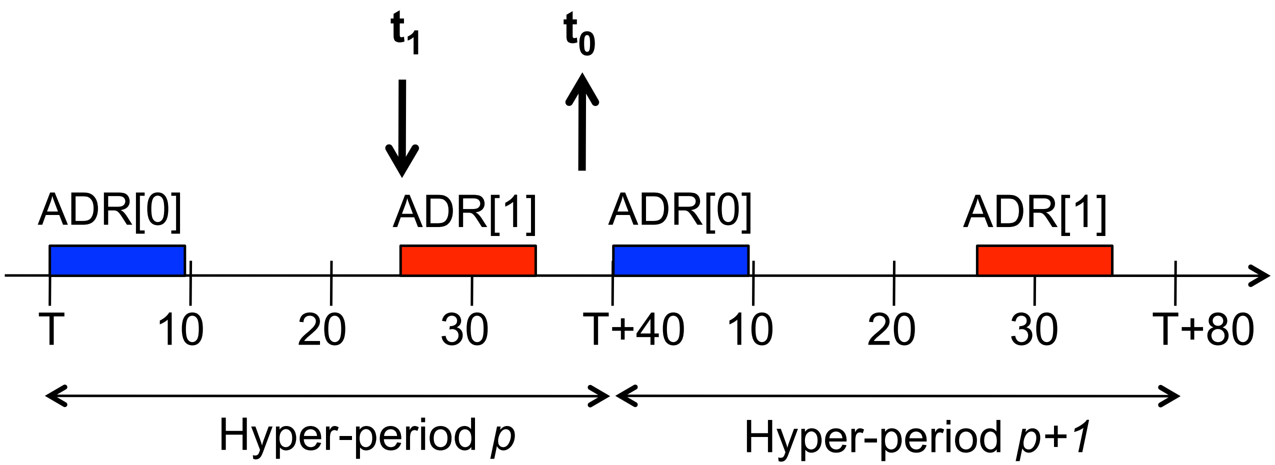
$$
\begin{array}{l}
O_{Aileron}, O_{RDC1}, O_{M1} \ldots \in \mathbb{R} \\
0 \leq O_{Aileron} \leq H \\
0 \leq O_{RDC1} \leq H \\
0 \leq O_{M1} \leq H \\
\ldots
\end{array}
$$

where H is the highest hyper-period of the system ($H = 40$ in the case-study)

Task modeling

- a task is characterized by a set of jobs in the hyper-period of its module

Example: *ADR* (2 jobs in $40ms$): $[0, 10]$, $[25, 35]$



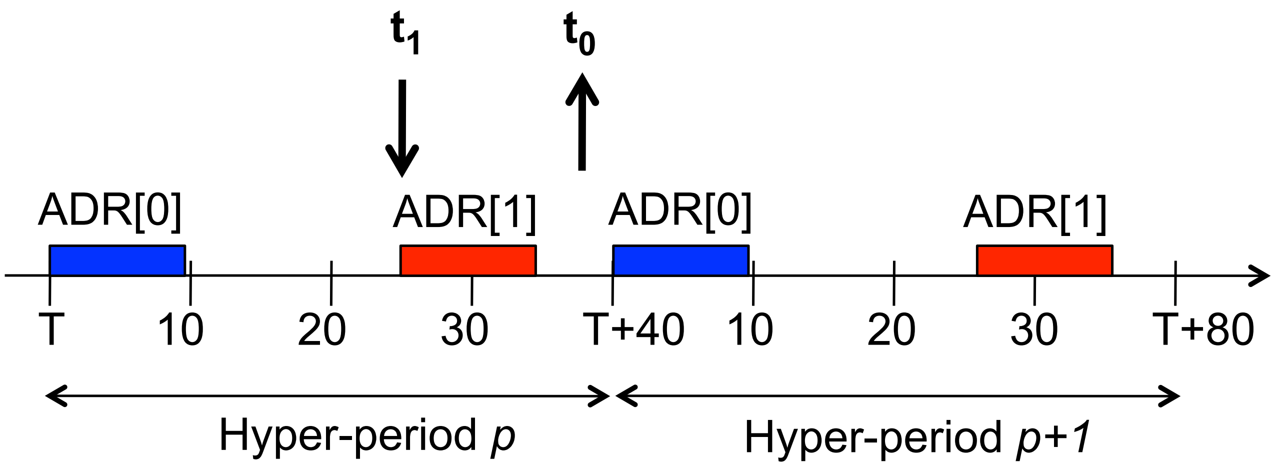
Problem:

- t_0 the date at which the output is observed
- what constraints to determine which job produces the output and the date t_1 from t_0?

Task modeling

- a task is characterized by a set of jobs in the hyper-period of its module

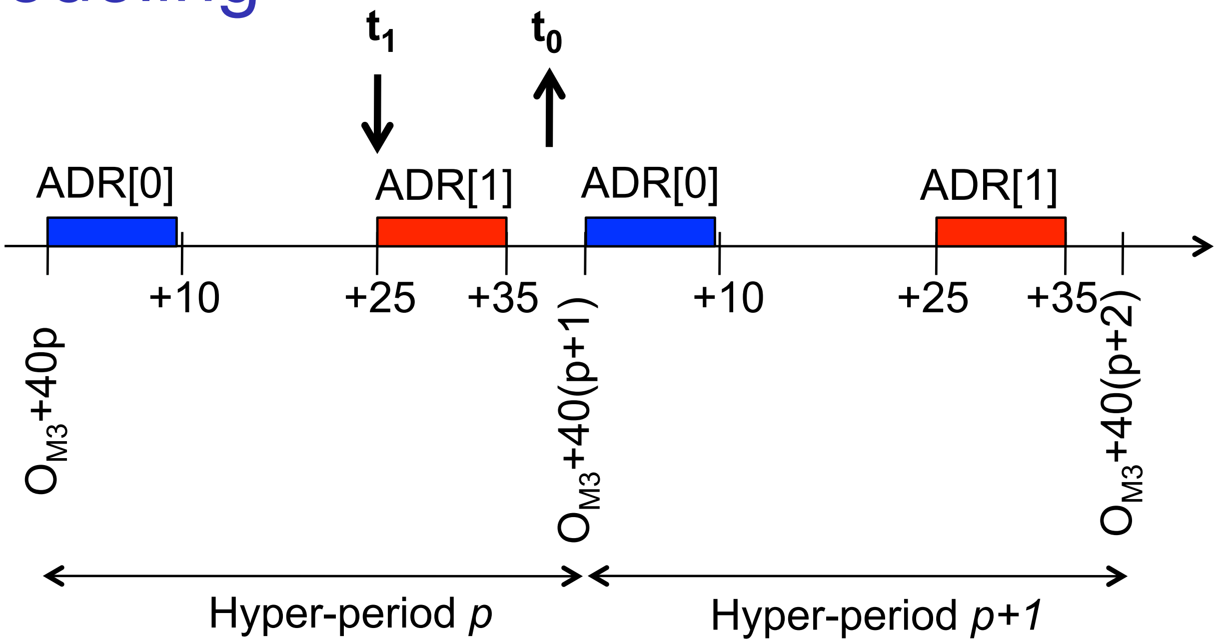
Example: *ADR* (2 jobs in 40ms): $[0, 10]$, $[25, 35]$



Problem:

- t_0 the date at which the output is observed
- what constraints to determine which job produces the output and the date t_1 from t_0?

Task modeling



Let:

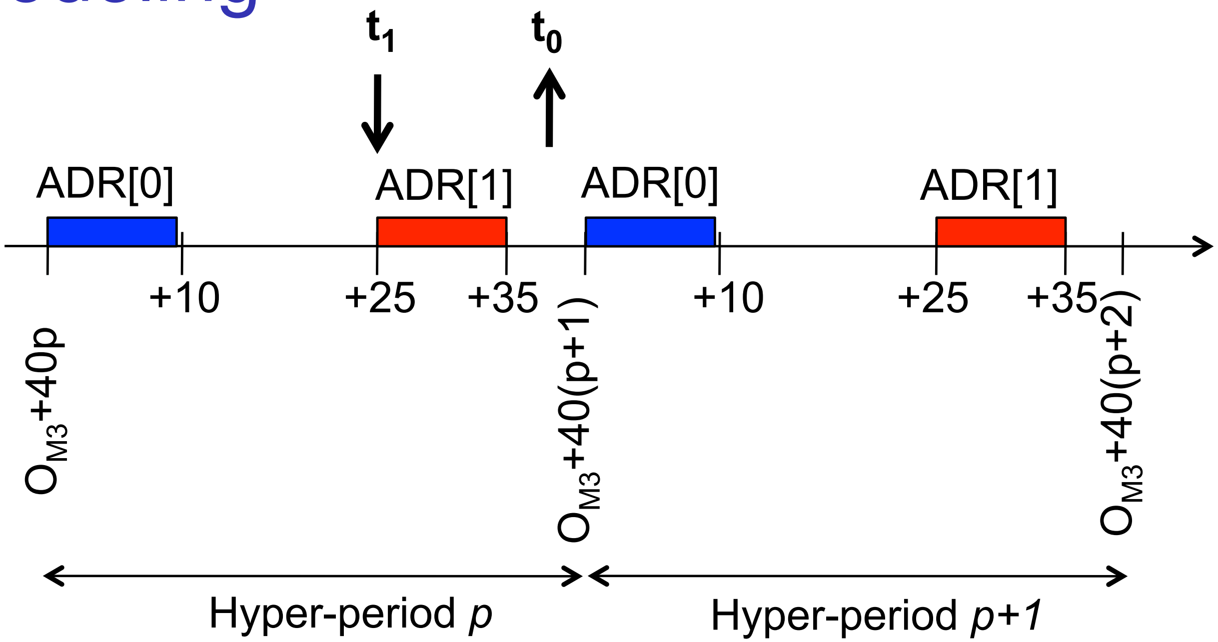
- $p \in \mathbb{N}$ (index of the current hyper-period at t_0)
- $B_{ADR[0]}, B_{ADR[1]} \in \{0, 1\}, B_{ADR[0]} + B_{ADR[1]} = 1$

Then:

$t_0 \geq O_{M3} + 40p + 0B_{ADR[0]} + 25B_{ADR[1]}$
$t_0 < O_{M3} + 40p + 35B_{ADR[0]} + (40 + 10)B_{ADR[1]}$
(recall: a job can produce its output at anytime in its time interval)

$t_1 = O_{M3} + 40p + 0B_{ADR[0]} + 25B_{ADR[1]}$

Task modeling



Let:

- $p \in \mathbb{N}$ (index of the current hyper-period at t_0)
- $B_{ADR[0]}, B_{ADR[1]} \in \{0, 1\}, B_{ADR[0]} + B_{ADR[1]} = 1$

Then:

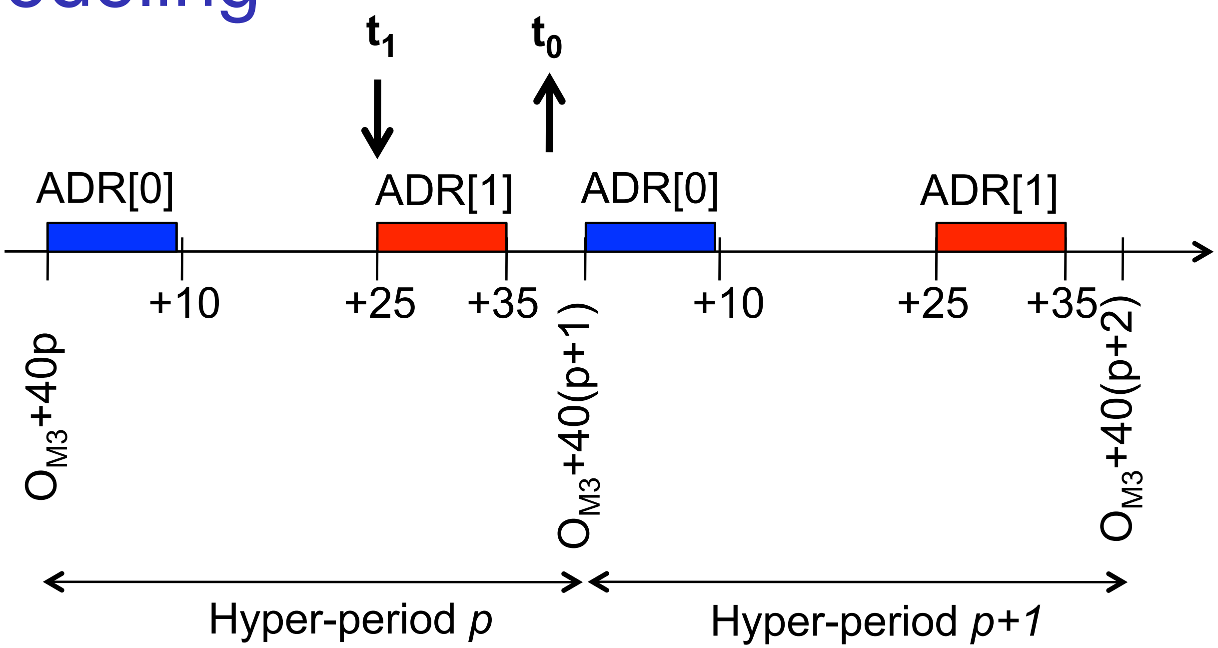
$t_0 \geq O_{M3} + 40p + 0B_{ADR[0]} + 25B_{ADR[1]}$

$t_0 < O_{M3} + 40p + 35B_{ADR[0]} + (40 + 10)B_{ADR[1]}$

(recall: a job can produce its output at anytime in its time interval)

$t_1 = O_{M3} + 40p + 0B_{ADR[0]} + 25B_{ADR[1]}$

Task modeling



Let:

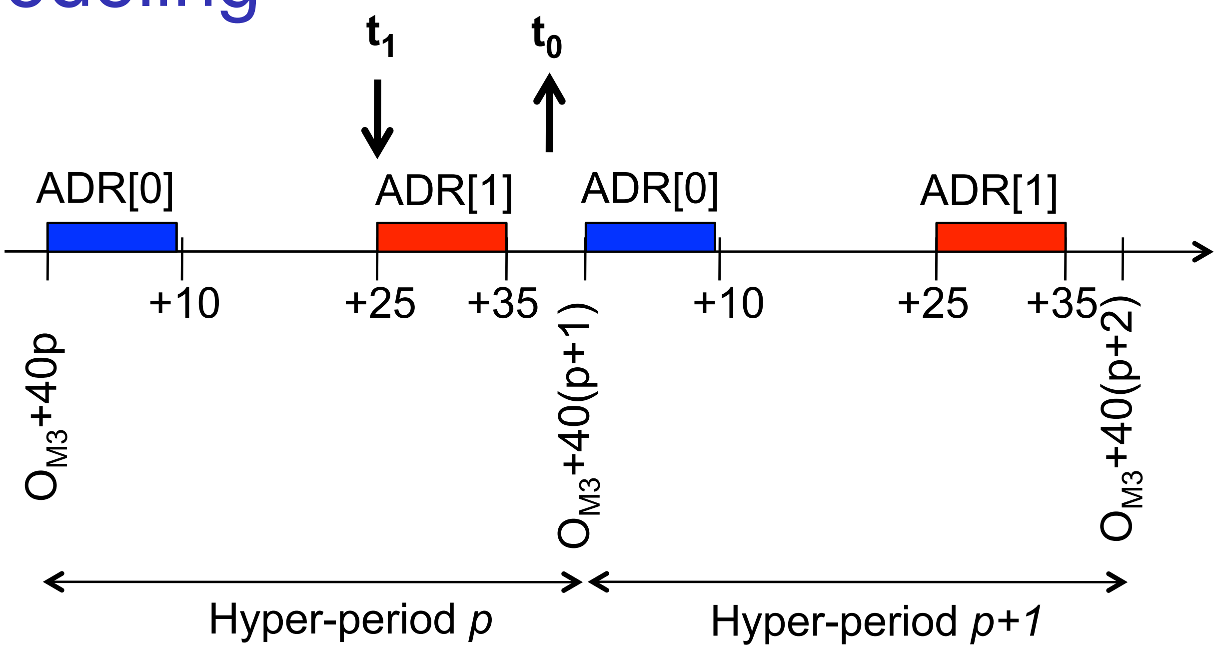
- $p \in \mathbb{N}$ (index of the current hyper-period at t_0)
- $B_{ADR[0]}, B_{ADR[1]} \in \{0, 1\}, B_{ADR[0]} + B_{ADR[1]} = 1$

Then:

$t_0 \geq O_{M3} + 40p + 0B_{ADR[0]} + 25B_{ADR[1]}$
$t_0 < O_{M3} + 40p + 35B_{ADR[0]} + (40 + 10)B_{ADR[1]}$
(recall: a job can produce its output at anytime in its time interval)

$t_1 = O_{M3} + 40p + 0B_{ADR[0]} + 25B_{ADR[1]}$

Task modeling



Let:

- $p \in \mathbb{N}$ (index of the current hyper-period at t_0)
- $B_{ADR[0]}, B_{ADR[1]} \in \{0, 1\}, B_{ADR[0]} + B_{ADR[1]} = 1$

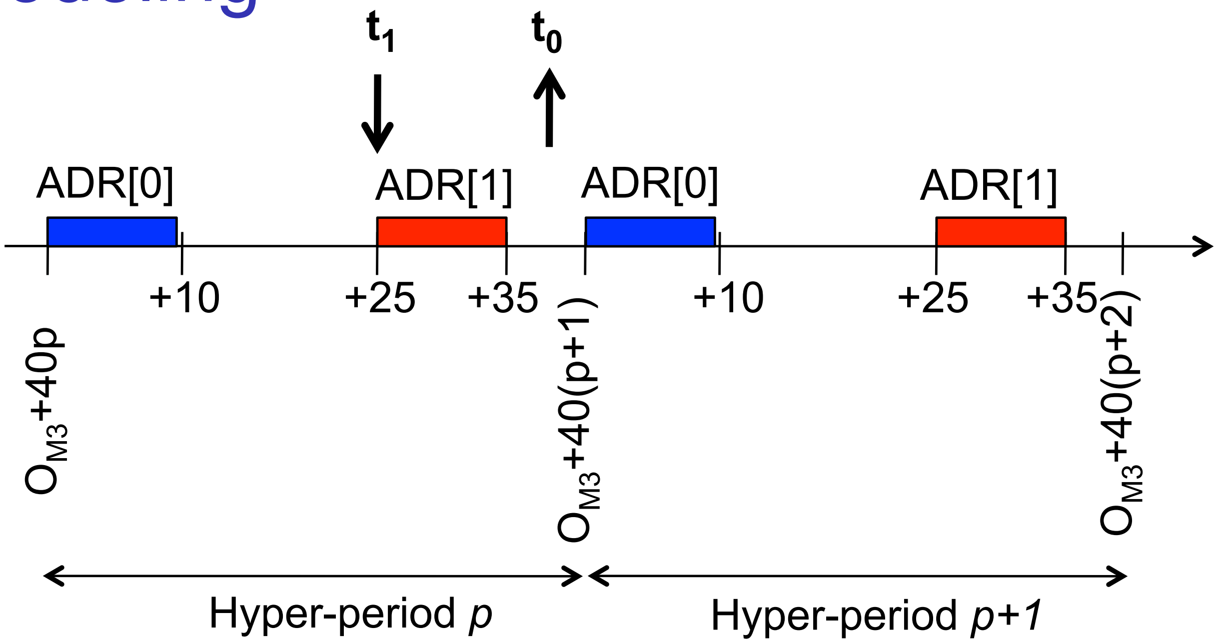
Then:

$$t_0 \geq O_{M3} + 40p + 0B_{ADR[0]} + 25B_{ADR[1]}$$
$$t_0 < O_{M3} + 40p + 35B_{ADR[0]} + (40 + 10)B_{ADR[1]}$$

(recall: a job can produce its output at anytime in its time interval)

$$t_1 = O_{M3} + 40p + 0B_{ADR[0]} + 25B_{ADR[1]}$$

Task modeling



Let:

- $p \in \mathbb{N}$ (index of the current hyper-period at t_0)
- $B_{ADR[0]}, B_{ADR[1]} \in \{0, 1\}, B_{ADR[0]} + B_{ADR[1]} = 1$

Then:

$$t_0 \geq O_{M3} + 40p + 0B_{ADR[0]} + 25B_{ADR[1]}$$
$$t_0 < O_{M3} + 40p + 35B_{ADR[0]} + (40 + 10)B_{ADR[1]}$$

(recall: a job can produce its output at anytime in its time interval)

$$t_1 = O_{M3} + 40p + 0B_{ADR[0]} + 25B_{ADR[1]}$$

Task modeling

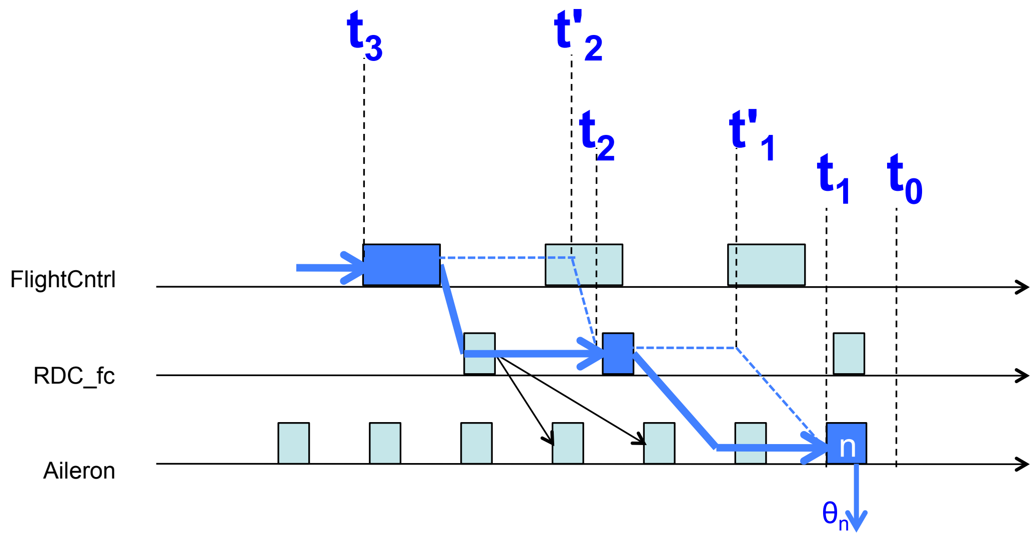
All solutions

$$(p, B_{ADR[0]}, B_{ADR[1]}, t_1) \in \mathbb{N} \times \{0, 1\} \times \{0, 1\} \times \mathbb{R}$$

satisfying the previous constraints characterize possible scenarios leading to the observation of the output at t_0.

Task modeling

Back to the case-study:

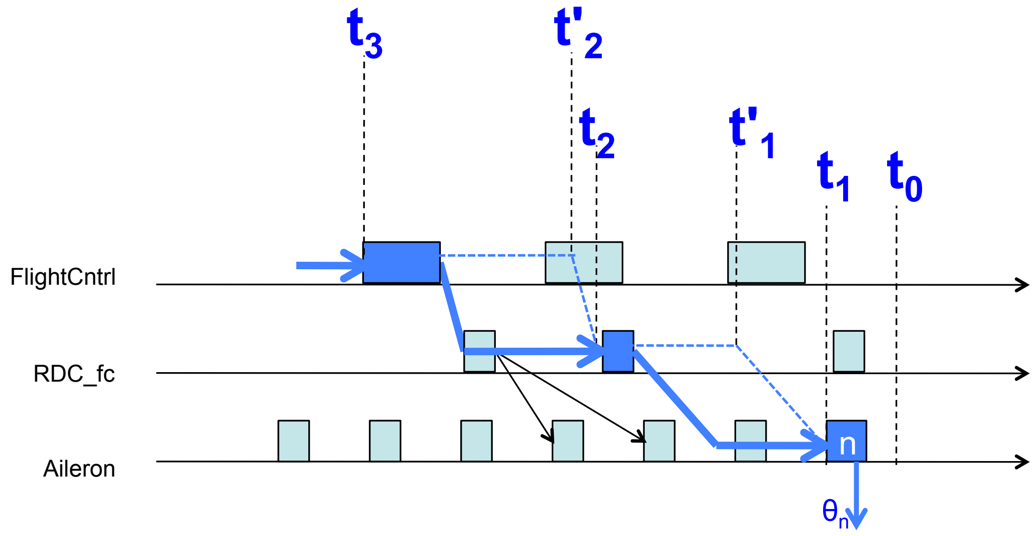


$t_0 \geq O_{Aileron} + 5p + 0B_{Aileron[0]}$
(recall: *Aileron* has only one job in its hyper-period (5ms))

$t_0 < O_{Aileron} + 5p + 6B_{Aileron[0]}$
(recall: the length of *Aileron* is 6ms)

$t_1 = O_{Aileron} + 5p + 0B_{Aileron[0]}$

Communication modeling



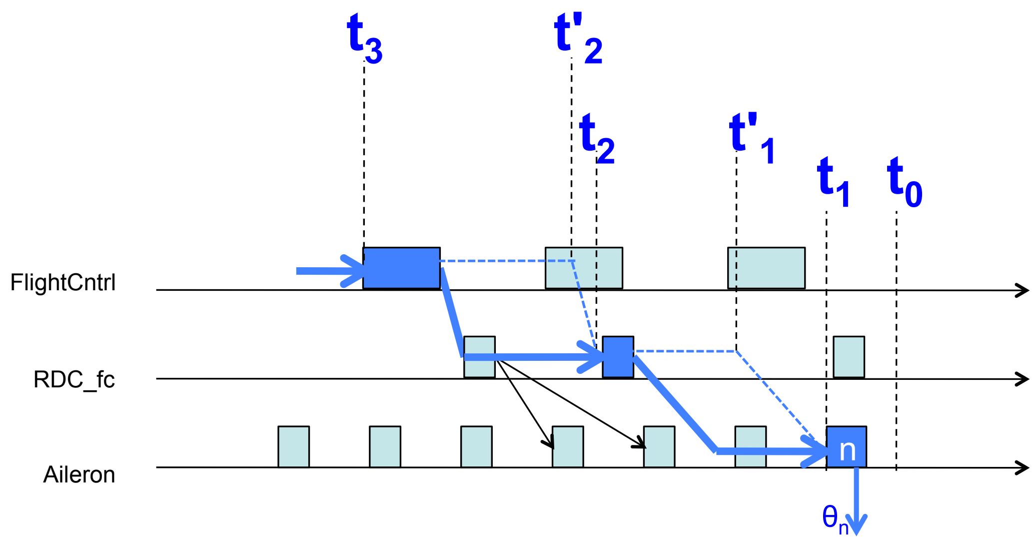
Let

- δ_{min} = the minimum communication time
- δ_{max} = the maximum communication time

$\Rightarrow$ Constraints:

$$t'_1 + \delta_{min} \leq t_1 \leq t'_1 + \delta_{max}$$

Communication modeling



Let

- δ_{min} = the minimum communication time
- δ_{max} = the maximum communication time

$\Rightarrow$ Constraints:

$$t'_1 + \delta_{min} \leq t_1 \leq t'_1 + \delta_{max}$$

Latency requirement modeling

And so on ... up to

$$t_8 = O_{Air\_sensor} + 5p_8 + 0B_{Air\_sensor[0]}$$

(t_8 is the date at which the total pressure corresponding to the Aileron position at t_0 is read; p_8 is the number of the corresponding hyper-period of the Air sensor)

$\Rightarrow$ The freshness expression:

$$F = t_0 - t_8$$

$\Rightarrow$ The worst case freshness is obtained on a particular behavior maximizing F.

$\Rightarrow$ Opimization problem:

maximize: F

Latency requirement modeling

And so on ... up to

$$t_8 = O_{Air\_sensor} + 5p_8 + 0B_{Air\_sensor[0]}$$

(t_8 is the date at which the total pressure corresponding to the Aileron position at t_0 is read; p_8 is the number of the corresponding hyper-period of the Air sensor)

$\Rightarrow$ The freshness expression:

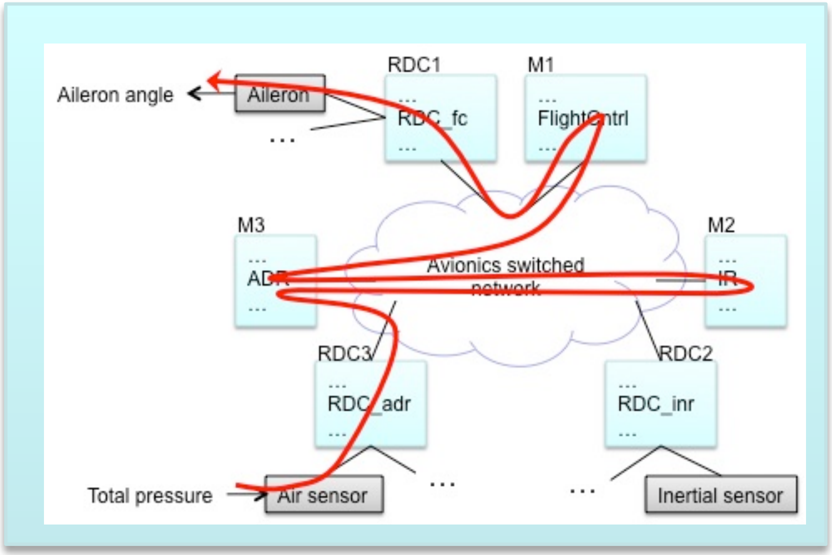
$$F = t_0 - t_8$$

$\Rightarrow$ The worst case freshness is obtained on a particular behavior maximizing F.

$\Rightarrow$ Opimization problem:

maximize: F

The tool chain



System model + WCF requirement

Encoding

//arrival in KC
$t_0 \leq O_1 + n_1 \cdot H_1 + 19Bkc_0 + 62Bkc_1 + 99Bkc_2$
$t_0 > O_1 + n_1 \cdot H_1 + 19Bkc_1 + 62Bkc_2 - Bkc_0 \cdot \mathit{Infty}$
$t_0 > O_1 + (n_1 - 1) \cdot H_1 + 99 - \mathit{Infty}(Bkc_0 - 1)$
$Bkc_0, Bkc_1, Bkc_2 \in \{0, 1\}$
$Bkc_0 + Bkc_1 + Bkc_2 = 1$

//output of KC
$t_1 = O_1 + n_1 \cdot H_1 + 31Bkc_0 + 74Bkc_1 + 11Bkc_2$

//timed channel from M_1 to M_2
$t_1 + 0.2 \leq t'_1 \leq t_1 + 2$

//arrival in $CockpitReqM$
$t'_1 \leq O_2 + n_2 \cdot H_2 + 31Bcr_0 + 76Bcr_1$
$t'_1 > O_2 + n_2 \cdot H_2 + 31Bcr_1 - Bcr_0 \cdot \mathit{Infty}$
$t'_1 > O_2 + (n_2 - 1) \cdot H_2 + 76 - \mathit{Infty}(Bcr_0 - 1)$
$Bcr_0, Bcr_1 \in \{0, 1\}$
$Bcr_0 + Bcr_1 = 1$

//output of $CockpitReqM$
$t_2 = O_2 + n_2 \cdot H_2 + 41Brc_0 + 101Brc_1$
...

MILP model

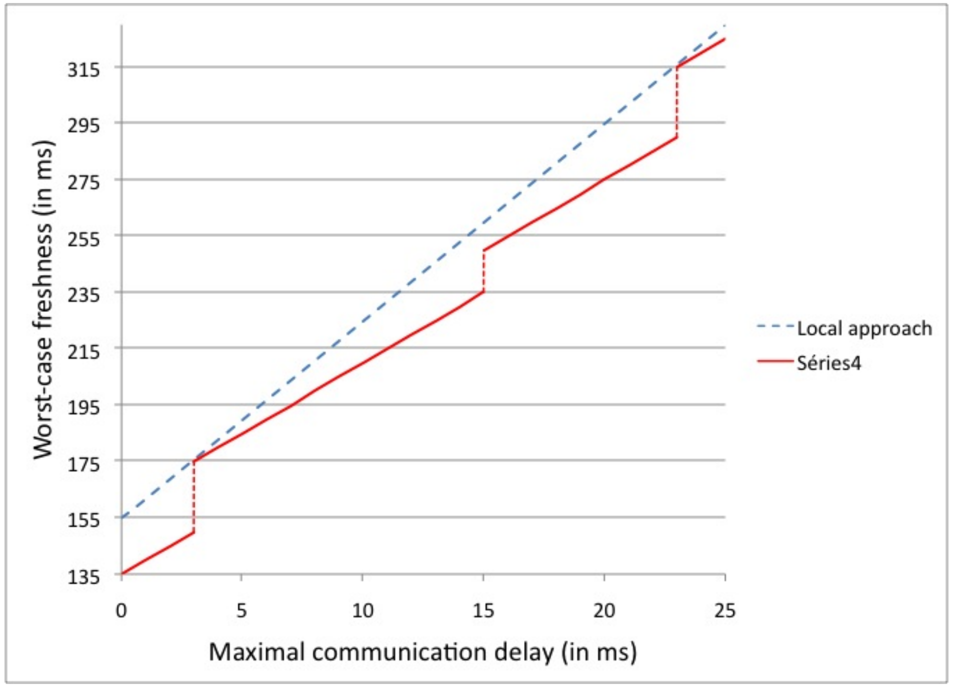
LP-solve

Worst-case latency +
a worst-case scenario

Outline

Results on the case study: global v.s. local approach

- global approach against the local one (i.e. sum of the local worst case)
- by varying the upper bound of the communication delays through the network



End-to-end freshness of aileron angle

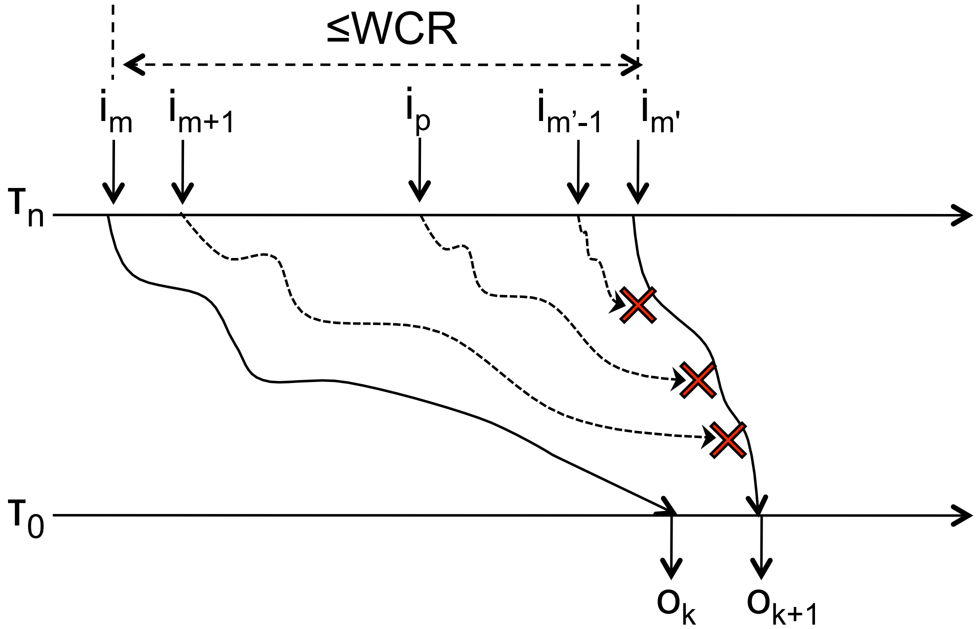
- red line = global approach
- dashed line = local approach

Outline

End-to-end reactivity requirement

End-to-end reactivity =

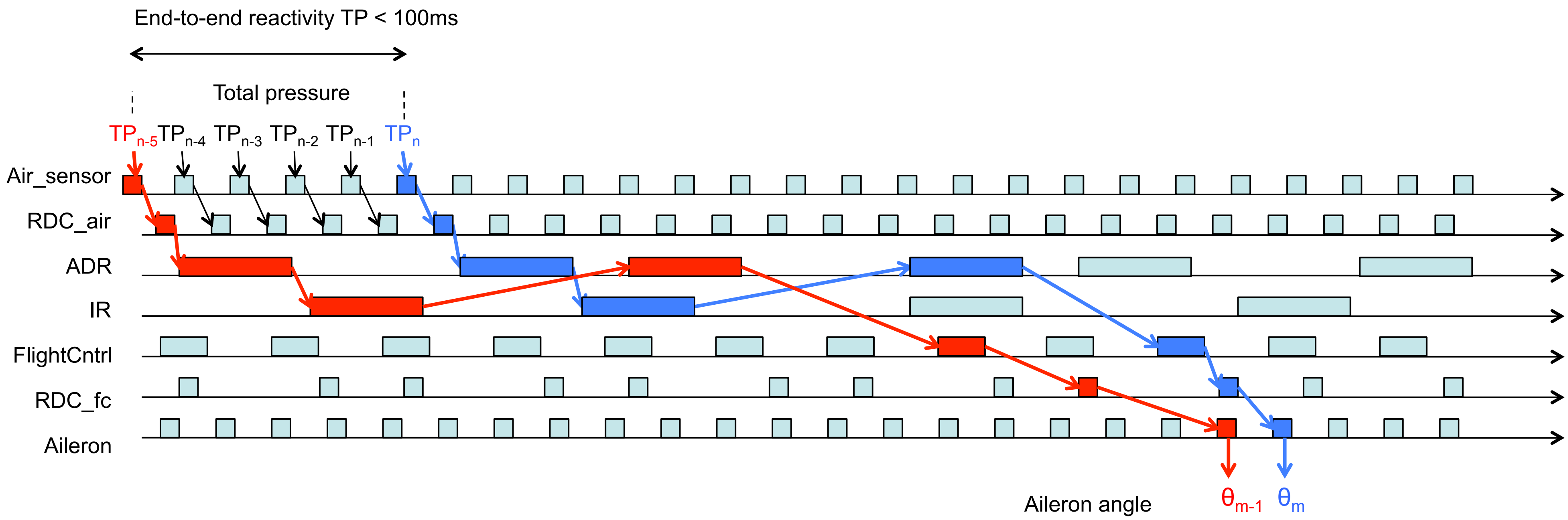
the minimal duration of an input signal, in order to be taken into account at the output of the chain



- i_m impacts o_k, $i_{m'}$ impacts o_{k+1}
- i_{m+1} to $i_{m'-1}$ do not impact the ouput (overwritten in the chain)

$\Rightarrow$ Worst Case Reactivity $\geq$ $date(i_{m'}) - date(i_m)$

End-to-end reactivity requirement

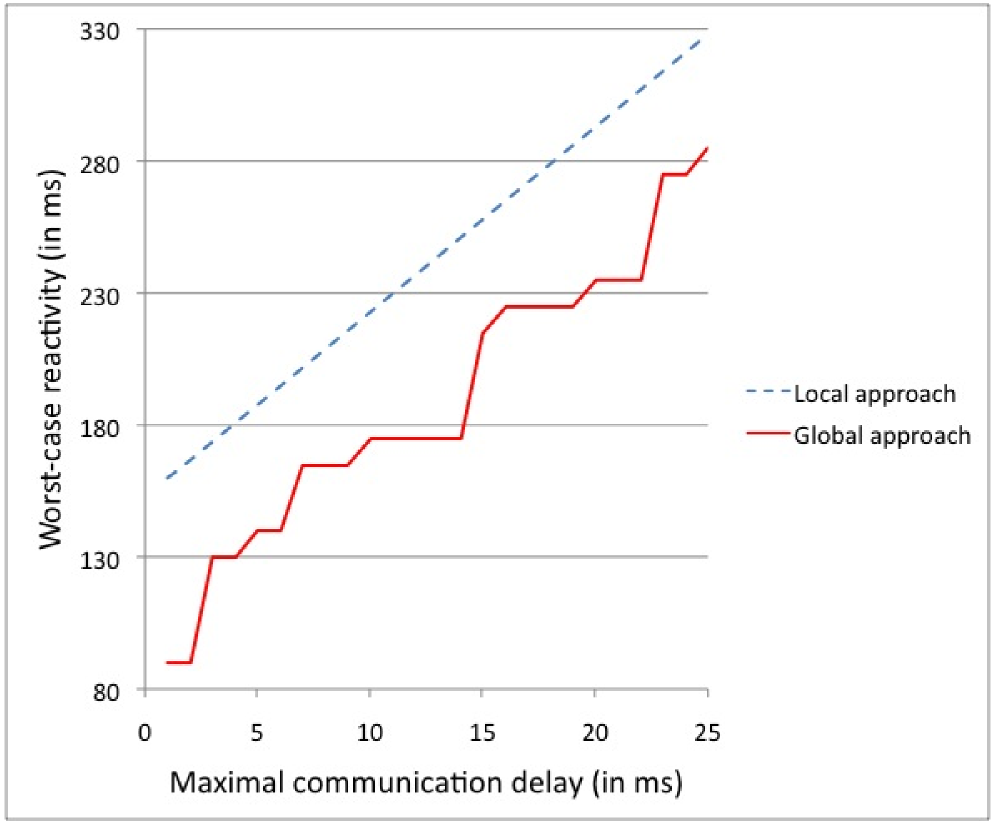


Approach:

- backtrack dependencies of two successive outputs
  - $\theta_m \rightsquigarrow TP_n$
  - $\theta_{m-1} \rightsquigarrow TP_{n'}$ (in the figure $n' = n - 5$)
- use the same contraints / variables as for freshness analysis
- maximize: $date(TP_n) - date(TP_{n'})$

Results on the case study: global v.s. local approach

- global approach against the local one (i.e. sum of the local worst case)
- by varying the upper bound of the communication delays through the network



- red line = global approach
- dashed line = local approach

End-to-end reactivity of total pressure

Outline

Conclusion and next work

Conclusion

- An efficient method for worst-case end-to-end freshness and reactivity analysis
- Extendable for best-case analysis
- Scalability:
  - takes less then 1 minute on the case-study
  - functional chain in realistic avionic systems contains at most 10 jumps from a task to another one (similar to the case-study)

  $\Rightarrow$ seems to be scalable

Next work

- Take into account the internal behavior of the tasks (i.e. latencies induced by the functional behavior of a task)

Thank you for your attention
(and many thanks to Pierre-Loïc Garoche)